**SOURCE CODE:**

solidity

// SPDX-License-Identifier: MIT

pragma solidity ^0.8.0;

// The FoodTrackingSystem contract

contract FoodTrackingSystem {

struct FoodItem {

uint256 id;

string name;

string origin;

uint256 timestamp;

}

mapping(uint256 => FoodItem) public foodItems;

uint256 public foodItemCount;

event FoodItemAdded(uint256 id, string name, string origin, uint256 timestamp);

// Function to add a new food item to the system

function addFoodItem(string memory \_name, string memory \_origin) public {

foodItemCount++;

foodItems[foodItemCount] = FoodItem(foodItemCount, \_name, \_origin, block.timestamp);

emit FoodItemAdded(foodItemCount, \_name, \_origin, block.timestamp);

}

// Function to get the details of a food item by its ID

function getFoodItem(uint256 id) public view returns (uint256, string memory, string memory, uint256) {

FoodItem memory foodItem = foodItems[id];

return (foodItem.id, foodItem.name, foodItem.origin, foodItem.timestamp);

}

}

This smart contract defines a simple FoodTrackingSystem where users can add food items with a name, origin, and a timestamp. They can also retrieve the details of a specific food item using its ID.

Please note that this is just a starting point and there are many features and enhancements that can be added to make the system more robust and secure. Additionally, you will need to develop a user interface and integrate it with the blockchain network using a suitable library like web3.js or ethers.js to interact with the smart contract.

It is highly recommended to consult with experienced blockchain developers or a development agency to ensure the successful implementation of a food tracking system based on blockchain technology.

**FRONT END:**

**<?xml version="1.0" encoding="UTF-8"?>**

**<projectDescription>**

**<name>FoodTrackingSystem</name>**

**<comment></comment>**

**<projects>**

**</projects>**

**<buildSpec>**

**<buildCommand>**

**<name>org.eclipse.ajdt.core.ajbuilder</name>**

**<arguments>**

**</arguments>**

**</buildCommand>**

**<buildCommand>**

**<name>org.eclipse.wst.common.project.facet.core.builder</name>**

**<arguments>**

**</arguments>**

**</buildCommand>**

**<buildCommand>**

**<name>org.springframework.ide.eclipse.core.springbuilder</name>**

**<arguments>**

**</arguments>**

**</buildCommand>**

**<buildCommand>**

**<name>org.eclipse.wst.jsdt.core.javascriptValidator</name>**

**<arguments>**

**</arguments>**

**</buildCommand>**

**<buildCommand>**

**<name>org.eclipse.wst.validation.validationbuilder</name>**

**<arguments>**

**</arguments>**

**</buildCommand>**

**<buildCommand>**

**<name>org.eclipse.m2e.core.maven2Builder</name>**

**<arguments>**

**</arguments>**

**</buildCommand>**

**</buildSpec>**

**<natures>**

**<nature>org.eclipse.ajdt.ui.ajnature</nature>**

**<nature>org.springframework.ide.eclipse.core.springnature</nature>**

**<nature>org.eclipse.jdt.core.javanature</nature>**

**<nature>org.eclipse.m2e.core.maven2Nature</nature>**

**<nature>org.eclipse.wst.common.project.facet.core.nature</nature>**

**<nature>org.eclipse.wst.common.modulecore.ModuleCoreNature</nature>**

**<nature>org.eclipse.wst.jsdt.core.jsNature</nature>**

**</natures>**